

Radon–Nikodym approximation in application to image reconstruction.

Vladislav Gennadievich Malyshkin*

Ioffe Institute, Politekhnicheskaya 26, St Petersburg, 194021, Russia

(Dated: November 3, 2015)

\$Id: RNimages.tex,v 1.105 2015/12/15 12:49:00 mal Exp \$

For an image pixel information can be converted to the moments of some basis Q_k , e.g. Fourier–Mellin, Zernike, monomials, etc. Given sufficient number of moments pixel information can be completely recovered, for insufficient number of moments only partial information can be recovered and the image reconstruction is, at best, of interpolatory type. Standard approach is to present interpolated value as a linear combination of basis functions, what is equivalent to least squares expansion. However, recent progress in numerical stability of moments estimation allows image information to be recovered from moments in a completely different manner, applying Radon–Nikodym type of expansion, what gives the result as a ratio of two quadratic forms of basis functions. In contrast with least squares the Radon–Nikodym approach has oscillation near the boundaries very much suppressed and does not diverge outside of basis support. While least squares theory operate with vectors $\langle fQ_k \rangle$, Radon–Nikodym theory operates with matrices $\langle fQ_jQ_k \rangle$, what make the approach much more suitable to image transforms and statistical property estimation.

* malyshki@ton.ioffe.ru

I. INTRODUCTION

Image information representation is a fundamental question of image processing and analysis. Most common basis is pixel basis. However given basis functions Q_k (e.g. Fourier, Zernike, orthogonal polynomials[1], etc.) image pixel information can be transformed to the moments of the basis[2–4]. Given sufficient number of moments a complete one-to-one mapping between pixel and moments information can be established. However, given limited number of moments a question arise: how image information can be recovered from moments available. Most common approach – representation of the result in a form of linear combination of basis functions, what is equivalent to least squares approximation. However, there is exist a different approach based on Radon–Nikodym derivatives[5] and its special case Nevai Operator[6, 7], where the result is represented as a ratio of two quadratic forms of basis functions. In contrast with least squares, which operate on vector moments $\langle fQ_k \rangle$ of observable value f , the Radon–Nikodym approach operate with matrices $\langle fQ_kQ_j \rangle$. Given recent progress in numerical stability of high order moments calculation[8] the matrices $\langle fQ_kQ_j \rangle$ can now be calculated without any difficulty to a very high order and Radon–Nikodym become practically applicable to image processing. This matrix approach, has a number of unique features, such as suppression of typical for least squares oscillations near the boundary and improved numerical stability. In addition to that the transition from vector to matrix allows many image transforms to be easily expressed in terms of matrix $\langle fQ_kQ_j \rangle$ transform and the approach allows to leverage matrix algebra in application to image processing.

II. BASIS EXPANSION

Consider some feature f (e.g. grayscale intensity), a basis $Q_k(x)$ (in 2D the basis would be $Q_{k_x}(x)Q_{k_y}(y)$) and the measure μ (in this paper the measure would be just the sum over the pixels). The moments are defined as:

$$\langle fQ_k \rangle = \int_{\text{supp}(\mu)} fQ_k d\mu \quad (1)$$

The Gramm matrix is defined by the basis and the measure:

$$G_{ij} = \langle Q_iQ_j \rangle \quad (2)$$

Then minimization of mean square difference between f and its approximation $A_{LS}(x)$ obtain standard least squares result:

$$A_{LS}(x) = \sum_{i,j} Q_i(x) (G^{-1})_{ij} < f Q_j > \quad (3)$$

Radon–Nikodym approximation $A_{RN}(x)$ can be obtained considering localized at x_0 states $\psi_{x_0}(x)$

$$\psi_{x_0}(x) = \frac{\sum_{i,j} Q_i(x) G_{ij}^{-1} Q_j(x_0)}{\sqrt{\sum_{i,j} Q_i(x_0) G_{ij}^{-1} Q_j(x_0)}} \quad (4)$$

and a form of Radon–Nikodym approximation, Nevai Operator[6, 7], then becomes:

$$A_{RN}(x) = \frac{\sum_{i,j,k,m} Q_i(x) G_{ij}^{-1} < f Q_j Q_k > G_{km}^{-1} Q_m(x)}{\sum_{i,j} Q_i(x) G_{ij}^{-1} Q_j(x)} \quad (5)$$

The main idea is to consider localized at x_0 states $\psi_{x_0}(x)$, which is related to delta-function expanded in $Q_k(x)$ basis with measure (1), and perform f reconstruction as $f(x_0) \approx \int dx f(x) \psi_{x_0}^2(x) / \int dx \psi_{x_0}^2(x)$. Important, that integration weight $\psi_{x_0}^2(x)$ is always positive what suppress oscillations typical for least squares, where the weight change sign. The $\psi_{x_0}(x)$ from (4) give exactly Nevai operator (5). For details and other $\psi_{x_0}(x)$ forms applicable for Radon–Nikodym estimation see Ref. [8]. The (5), while is very different from least squares in concept, uses, nevertheless, almost the same input: Gramm matrix inverse G^{-1} and $< Q_i Q_k f >$ matrix obtained from $< Q_j f >$ moments. The (5) is a ratio of two polynomial functions. It was shown in Ref. [9] that in multi-dimensional signal processing stable estimators can be only of two quadratic forms ratio and the (5) is exactly of this form.

Let us apply least squares and Radon–Nikodym expressions to some real life cases.

A. 1D Example: Runge Function.

Before we start considering 2D images, let us start with simple 1D example, take Runge function

$$f(x) = \frac{1}{1 + 25x^2} \quad (6)$$

Using the measure

$$< f Q_k > = \int_{-1}^1 f(x) Q_k(x) dx \quad (7)$$

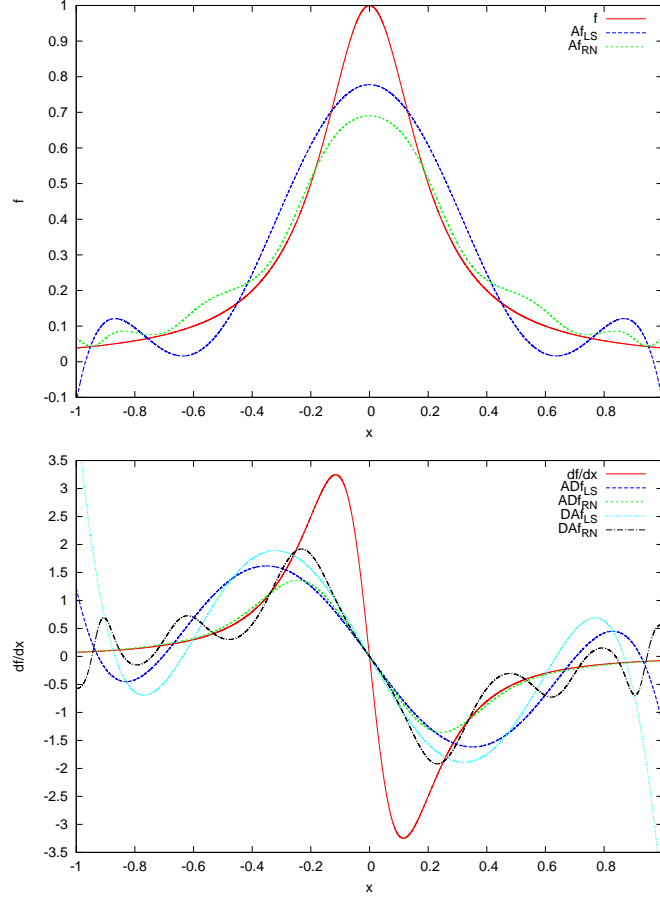


FIG. 1. Top chart: Runge function f , least squares approximation Af_{LS} and Radon–Nikodym approximation Af_{RN} . Bottom chart: Runge function derivative $\frac{df}{dx}$, least squares approximation ADf_{LS} and Radon–Nikodym approximation ADf_{RN} of the derivative $\frac{df}{dx}$. Differentiated interpolations of f for least squares (DAf_{LS}) and Radon–Nikodym (DAf_{RN}), that should not be used in applications are also presented as an example.

and the basis, for numerical stability of calculations, is chosen as Legendre polynomials $Q_k(x) = P_k(x)$ (given the measure all polynomial bases provide identical results, but numerical stability of calculations is different).

The $A_{RN}(x)$ calculation algorithm is this: given N elements in basis using (7) definition calculate vector moments $\langle fQ_m \rangle$ and $\langle Q_m \rangle$ for $m = [0 \dots 2N - 1]$. Then, applying polynomials multiplication operation, for $j, k = [0 \dots N - 1]$ obtain matrix moments $\langle fQ_jQ_k \rangle$ and $G = \langle Q_jQ_k \rangle$ to be used in Eq. (3) and (5).

In top chart of Fig. 1 least squares and Radon–Nikodym interpolations are presented for $N = 7$ and the measure (7). One can see that near edges oscillations are much less severe,

when Radon–Nikodym approximation as polynomials ratio is used for the interpolation of f . The major behavior differences for least square and Radon–Nikodym approximations are: Least squares have diverging oscillations near measure support boundaries and tend to infinity with the distance to measure support increase. Radon–Nikodym have converging oscillations near measure support boundaries and tend to a constant with the distance to measure support increase.

Another, worth to mention point, is related to derivatives calculation. For this the moments $\langle Q_k \frac{df}{dx} \rangle$ should be calculated first, and only then applying Radon–Nikodym approximation like (5) using derivative moments. If one, instead of using the $\langle Q_k \frac{df}{dx} \rangle$ moments, would differentiate f approximation expression (5) directly – the result will be incorrect. To illustrate the point in bottom chart of Fig. 1 least squares and Radon–Nikodym interpolations of Runge function derivative are presented for $N = 7$ and measure (7). The differentiated approximations of Runge function are also presented.

The code calculating this 1D example is available[10], see the ExampleRungeFunction.scala file.

B. 2D example: Lena image.

Let us consider 2D case of image grayscale intensity interpolation. Because the example is illustrative, let us take a sum over image pixels as the measure. As a basis, in principle, monomials $x^{k_x} y^{k_y}$ can be used, but for numerical stability reasons the basis should be chosen as orthogonal functions with respect to some measure. For an image of d_x on d_y pixels the moments of pixel–dependent grayscale intensity $f(t_x, t_y)$ (here $t_x = [0 \dots d_x - 1]$ and $t_y = [0 \dots d_y - 1]$ index pixel number) are:

$$\begin{aligned} < f Q_{k_x, k_y} > = \\ & \sum_{t_x=0}^{d_x-1} \sum_{t_y=0}^{d_y-1} f(t_x, t_y) Q_{k_x}(t_x/(d_x-1)) Q_{k_y}(t_y/(d_y-1)) \end{aligned} \quad (8)$$

In the Eq. (8) the basis can be chosen as Legendre or Chebyshev polynomials shifted to $[0 \dots 1]$ interval: $Q_k(x) = P_k(2x - 1)$ and the argument of Q is pixel coordinate converted to this interval: $x = t_x/(d_x - 1)$ and $y = t_y/(d_y - 1)$. When $d_x \rightarrow \infty$ $d_y \rightarrow \infty$ the Gramm matrix $< Q_{i_x, i_y} Q_{j_x, j_y} >$ is diagonal because of Legendre polynomials orthogonality and trivially invertable, but we used sample–calculated matrix, because the d_x and d_y can be rather small

or when the basis is chosen as Chebyshev polynomials $Q_k(x) = T_k(2x - 1)$. Note, that for a given measure all polynomial bases (e.g. Legendre, Chebyshev, monomials) give identical results, but numerical stability of calculations is drastically different, because Gramm matrix condition number depend strongly on basis choice[11]. For successful application in image reconstrction Chebyshev moments see Ref. [12] and for Legendre moments see Ref. [13].

The numerical library we developed, see[8] Appendix A, is able to manipulate polynomials in Chebyshev, Legendre, Laguerre and Hermite bases directly, what allows a stable basis to be used and calculate the moments to a very high order. Numerical calculations with polynomials in general basis were introduced in [14] and similar technique was used in [15] for Gauss quadratures calculation in Chebyshev basis. In this paper we used general polynomial basis approach (to achieve numerical stability) and applied it to Radon–Nikodym approximation calculation.

The A_{RN} calculation algorithm is this: given image size d_x and d_y and basis dimension N_x and N_y using (8) definition calculate vector moments $\langle f Q_{m_x, m_y} \rangle$ and $\langle Q_{m_x, m_y} \rangle$ for $m_x = [0 \dots 2N_x - 1]$ and $m_y = [0 \dots 2N_y - 1]$. Then, applying polynomials multiplication operation, for $j_x, k_x = [0 \dots N_x - 1]$ and $j_y, k_y = [0 \dots N_y - 1]$ obtain matrix moments $\langle f Q_{j_x, j_y} Q_{k_x, k_y} \rangle$ and $G = \langle Q_{j_x, j_y} Q_{k_x, k_y} \rangle$ to be used in Eq. (3) and (5).

In Fig. 2 we present original 512x512 ($d_x = d_y = 512$) grayscale Lena image, then for $N_x = N_y = 50$ apply least squares (3) and Radon–Nikodym (5) transforms. Same calculations, but for $N_x = N_y = 100$ are presented in Fig. 3. (The calculations for $N_x = N_y = 100$ are rather slow, because we did not use any optimization, but the point of the paper is to demonstrate practical applicability of Radon–Nikodym type of interpolation and stability of high order moments calculation when a stable basis is chosen.)

The least squares interpolation, same as in 1D case, present typical for least squares intensity oscillations near image edges, while Radon–Nikodym has these oscillations very much supressed. Another important feature of Radon–Nikodym is that it preserves the sign of interpolated function, i.e. the grayscale intencity f never become negative, what may happen easily for least squares. The code calculating this example is available[10], see the ExampleImageInterpolation.scala file. The calculations have been performed in both: Legendre and Chebyshev bases. For $N_x = N_y = 50$ the results are identical, when interpolated grayscale is converted back to 1-byte values. For $N_x = N_y = 100$ the results in two bases are almost identical (indistinguishable visually), but testing show that in Legendre



FIG. 2. Original Lena image(top left), and for $N_x = N_y = 50$ least squares(top right) and Radon-Nikodym (bottom). PNG originals are available from [16].

basis numerical instabilities just started to show up in multiplication operation, because of factorial-like coefficients in $P_n(x)P_m(x)$ expansion. One can expect more instability in Legendre basis at $N_x = N_y > 100$ (note, that for given N we calculate $0..2N-1$ moments). In this sense Chebyshev multiplication $T_n(x)T_m(x) = \frac{1}{2}T_{n+m}(x) + \frac{1}{2}T_{|n-m|}(x)$ is special because the coefficients of product expansion do not grow or vanish for large $n; m$, so Chebyshev products can be stably calculated to a very high order and in the same time for discrete measures the Gramm matrix (2) possess a good condition number[11] in this basis.



FIG. 3. Interpolation of Lena image for $N_x = N_y = 100$. Least squares(left) and Radon-Nikodym (right). PNG originals are available from [16].

III. DISCUSSION AND NATURAL BASIS

In this paper we present a novel approach to image restoration from moments: the result is of Radon–Nikodym type where the result is a ratio of two quadratic forms of basis functions, and, in case of polynomial bases, is just two polynomials ratio. This approach, is based on matrices, not on vectors, what make calculations significantly more stable. In a way how Radon–Nikodym approach improved interpolation of a function, the transition from a vector $\langle Q_k f \rangle$ to matrix $\langle Q_j Q_k f \rangle$ can similarly improve calculations of image properties, expressible through averages. Define an average \overline{f} as:

$$\begin{aligned} \overline{f} &= \frac{\text{Spur} \left(\sum_k G_{jk}^{-1} \langle Q_k f Q_l \rangle \right)}{\dim G} \\ &= \frac{\sum_{j,k} G_{jk}^{-1} \langle Q_k f Q_j \rangle}{\dim G} \end{aligned} \quad (9)$$

where Spur is matrix trace (sum of diagonal elements) operator. The (9) definition can be also applied to estimation of an average of products, i.e.

$$\overline{fg} = \frac{\sum_{j,k,m,i} G_{jk}^{-1} \langle Q_k f Q_m \rangle G_{mi}^{-1} \langle Q_i g Q_j \rangle}{\dim G} \quad (10)$$

What allows image features cross-correlation to be expressed as matrix Spur. An important feature of the approach is that many image transforms can be easily expressed as a transform

of matrix $\langle Q_k f Q_m \rangle$, what makes proposed matrix approach extremely practical, when a stable basis is chosen. Numerical library providing four stable bases (Legendre Chebyshev, Laguerre, Hermite) is available from author[10].

And in conclusion we want to mention that generalized eigenvalues problem

$$\sum_m \langle Q_k f Q_m \rangle \psi_m^{(s)} = \lambda^{(s)} \sum_m \langle Q_k Q_m \rangle \psi_m^{(s)} \quad (11)$$

$$\psi^{(s)}(x) = \sum_m \psi_m^{(s)} Q_m(x) \quad (12)$$

when solved[17] provide a “natural basis” of eigenvectors $\psi^{(s)}$ in which both matrices $\langle Q_k f Q_m \rangle$ and $\langle Q_k Q_m \rangle$ are simultaneously diagonal. Besides providing exceptional numerical stability this basis is a “natural basis” for the image, and can be extremely convenient to store and process image information. For example, because

$$\langle \psi^{(r)} \psi^{(s)} \rangle = \delta_{rs} = G_{rs} \quad (13)$$

$$\langle \psi^{(r)} f \psi^{(s)} \rangle = \lambda^{(s)} \delta_{rs} \quad (14)$$

the Gramm matrix is diagonal in natural basis — the cross-correlation of image features (10), calculated as matrix Spur, take exceptionally simple form. This “natural basis” can be considered as Radon–Nikodym derivatives generalization. While Radon–Nikodym derivatives are based on localized at x_0 states $\psi_{x_0}(x)$ from (4) the eigenfunctions $\psi^{(s)}$ from (12) have no such localization constrain and their localization depend only on image properties. The value of this “generalized Radon–Nikodym derivative” at $\psi^{(s)}$ state is the eigenvalue $\lambda^{(s)} = \langle \psi^{(s)} f \psi^{(s)} \rangle / \langle \psi^{(s)} \psi^{(s)} \rangle$. The difference between Radon–Nikodym and “generalized Radon–Nikodym” is similar to conceptual difference[5] between Riemann integral, where the terms are grouped by their closeness in argument–space, like $\psi_{x_0}(x)$ from (4), and Lebesgue integral where the terms are grouped by their closeness in value–space, like $\psi^{(s)}$ from (12). A Lebesgue–type integration using “generalized Radon–Nikodym” would look, schematically, like this: For the f , defining Lebesgue integration, solve the (11) problem. Split interval of f values to a number of $[f_i; f_i + df]$ intervals. Then define Lebesgue measure $d\mu$: for every such interval count the number of eigenvalues $\lambda^{(s)}$ that fall within interval range $f_i \leq \lambda^{(s)} < f_i + df$, this number would be the Lebesgue measure μ_i . Then Lebesgue integral of some function $\int g(f) d\mu$ is just $\sum_i g(f_i) \mu_i$. The concept is very similar to the “density of states” concept from quantum mechanics, where the density of states

is a number of Hamiltonian eigenvalues that fall within given energy interval. In practice the Lebesgue-type integration is most often performed in pixel basis, where the number of pixels with f falling within interval range $f_i \leq f < f_i + df$ is considered to be the Lebesgue measure μ_i . When Lebesgue-type integration is performed in “natural basis” the number of eigenvalues, instead of the number of pixels, is considered to be the Lebesgue measure μ_i .

-
- [1] Vilmos Totik, “Orthogonal polynomials,” *Surveys in Approximation Theory* **1**, 70–125 (11 Nov. 2005).
 - [2] Ramakrishnan Mukundan and KR Ramakrishnan, *Moment functions in image analysis: theory and applications*, Vol. 100 (World Scientific, 1998).
 - [3] Jean-Charles Pinoli, *Mathematical Foundations of Image Processing and Analysis*, Vol. 1 (John Wiley & Sons, 2014).
 - [4] Barmak Honarvar, Raveendran Paramesran, and Chern-Loon Lim, “Image reconstruction from a complete set of geometric and complex moments,” *Signal Processing* **98**, 224–232 (2014).
 - [5] A. N. Kolmogorov and S. V. Fomin, *Elements of the Theory of Functions and Functional Analysis* (Martino Fine Books (May 8, 2012), 8 May 2012).
 - [6] Paul G Nevai, “Orthogonal polynomials.” *Memoirs of the American Mathematical Society* **213** (1979).
 - [7] Paul G Nevai, “Géza Freud, Orthogonal Polynomials. Christoffel Functions. A Case Study,” *Journal Of Approximation Theory* **48**, 3–167 (1986).
 - [8] Vladislav Gennadievich Malyshkin and Ray Bakhramov, “Mathematical Foundations of Real-time Equity Trading. Liquidity Deficit and Market Dynamics. Automated Trading Machines. <http://arxiv.org/abs/1510.05510>,” *ArXiv e-prints* (2015), arXiv:1510.05510 [q-fin.CP].
 - [9] Gennadii Stepanovich Malyshkin, *Optimal and Adaptive Methods of Hydroacoustic Signal Processing. Vol 1. Optimal methods. (in Russian)*. (Elektropribor Publishing, 2009).
 - [10] Vladislav Gennadievich Malyshkin, (2014), the code for polynomials calculation, <http://www.ioffe.ru/LNEPS/malyshkin/code.html>.
 - [11] Bernhard Beckermann, *On the numerical condition of polynomial bases: estimates for the condition number of Vandermonde, Krylov and Hankel matrices*, Ph.D. thesis, Habilitationsschrift, Universität Hannover (1996).

- [12] R Mukundan, SH Ong, and Poh Aun Lee, “Image analysis by Tchebichef moments,” *Image Processing, IEEE Transactions on* **10**, 1357–1364 (2001).
- [13] Amy Chiang and Simon Liao, “Image analysis with Legendre moment descriptors,” *Journal of Computer Science* **11**, 127–136 (2014).
- [14] John Maroulas and Stephen Barnett, “Polynomials With Respect to a General Basis. II. Applications,” *Journal of Mathematical Analysis Applications* **72**, 599–614 (1979).
- [15] Dirk P Laurie and Laurette Rolfes, “Computation of Gaussian quadrature rules from modified moments,” *Journal of Computational and Applied Mathematics* **5**, 235–243 (1979).
- [16] Vladislav Gennadievich Malyshkin, “Radon–Nikodym approximation in application to image analysis. <http://arxiv.org/abs/1511.01887>,” *ArXiv e-prints* (2015), arXiv:1511.01887 [cs.CV].
- [17] In general case generalized eigenvector problem, when scalar product is defined not by a unit matrix, but by some other positively defined matrix, Gramm matrix in our case, is not any more problematic to solve numerically, than regular eigenvalues problem. It can be solved using standard, e.g. LAPACK[18] routines dsygv, dsygvd and similar.
- [18] “Lapack version 3.5.0,” (2013).

This figure "lena_LS_50x50.png" is available in "png" format from:

<http://arxiv.org/ps/1511.01887v2>

This figure "lena_RN_50x50.png" is available in "png" format from:

<http://arxiv.org/ps/1511.01887v2>

This figure "lena_LS_100x100.png" is available in "png" format from:

<http://arxiv.org/ps/1511.01887v2>

This figure "lena_RN_100x100.png" is available in "png" format from:

<http://arxiv.org/ps/1511.01887v2>

This figure "lena_orig.png" is available in "png" format from:

<http://arxiv.org/ps/1511.01887v2>